

AMENDMENTS TO THE CLAIMS

Please amend the claims. The following listing of claims replaces all previous versions in the Application:

What is claimed is:

1-27. (Canceled)

28. (Currently Amended) A method comprising:

determining if a computer, using firmware but no additional software, cannot read a block of data on a medium coupled with the computer during a pre-boot phase; and
in a case where the computer, using firmware but no additional software, cannot read the block of data,
reading, using firmware but no additional software, a header on the medium that describes a location of a program on the medium, the program to enable the computer to read the block of data, and
loading the program into the computer.

29. (Previously Presented) The method of claim 28 wherein the program is a file system driver.

30. (Previously Presented) The method of claim 29 wherein the file system driver is to operate in accordance with the Extensible Firmware Interface (EFI) framework standard.

31. (Previously Presented) The method of claim 29 further comprising:
mounting a file system using the file system driver; and
reading the data using the file system.

32. (Previously Presented) The method of claim 28 wherein the program is a pre-boot recovery utility.

33. (Previously Presented) The method of claim 32 wherein the pre-boot recovery utility is an Extensible Firmware Interface (EFI) application.

34. (Previously Presented) The method of claim 32 further comprising recovering a storage device coupled with the computer by reading a portion of the block of data and writing the portion to the storage device using the pre-boot recovery utility.

35. (Previously Presented) The method of claim 32 further comprising recovering a corrupted operating system boot target stored on the storage device using the pre-boot recovery utility, wherein the medium includes a magnetic backup tape.

36. (Currently Amended) An article of manufacture comprising:
a first machine-readable medium including a plurality of instructions which when executed perform operations comprising:
determining if a computer, using firmware but no additional software,
cannot read a block of data on a medium coupled with the computer during a pre-boot phase; and
in a case where the computer, using firmware but no additional software, cannot read the block of data,
reading, using firmware but no additional software, a header on the medium that describes a location of a program on the medium, the program to enable the computer to read the block of data, and loading the program into the computer.

37. (Previously Presented) The article of manufacture of claim 36 wherein the program is a file system driver.

38. (Previously Presented) The article of manufacture of claim 37 wherein the file system driver is to operate in accordance with the Extensible Firmware Interface (EFI) framework standard.

39. (Previously Presented) The article of manufacture of claim 37 wherein execution of the plurality of instructions further perform operations comprising:

mounting a file system using the file system driver; and
reading the data using the file system.

40. (Previously Presented) The article of manufacture of claim 36 wherein the program is a pre-boot recovery utility.

41. (Previously Presented) The article of manufacture of claim 40 wherein the pre-boot recovery utility is an Extensible Firmware Interface (EFI) application.

42. (Previously Presented) The article of manufacture of claim 40 wherein execution of the plurality of instructions further perform operations comprising recovering a storage device coupled with the computer by reading a portion of the block of data and writing the portion to the storage device using the pre-boot recovery utility.

43. **(Previously Presented)** The article of manufacture of claim 40 further comprising recovering a corrupted operating system boot target stored on the storage device using the pre-boot recovery utility, wherein the second medium includes a magnetic backup tape.

44. **(Currently Amended)** A computer system, comprising:

a processor; and

at least one non-volatile storage device operatively coupled to the processor, the

at least one non-volatile storage device including firmware instructions which when executed by the processor perform operations comprising:

determining if a computer, using firmware but no additional software,

cannot read a block of data on a medium coupled with the computer during a pre-boot phase; and

in a case where the computer, using firmware but no additional software, cannot read the block of data,

reading, using firmware but no additional software, a header on the medium that describes a location of a program on the medium, the program to enable the computer to read the block of data, and loading the program into the computer.

45. **(Previously Presented)** The system of claim 44 wherein the program is a file system driver.

46. **(Previously Presented)** The system of claim 45 wherein the file system driver is to operate in accordance with the Extensible Firmware Interface (EFI) framework standard.

47. (Previously Presented) The system of claim 45 wherein execution of the firmware instructions further perform operations comprising:

mounting a file system using the file system driver; and
reading the data using the file system.

48. (Previously Presented) The system of claim 44 wherein the program is a pre-boot recovery utility.

49. (Previously Presented) The system of claim 48 wherein the pre-boot recovery utility is an Extensible Firmware Interface (EFI) application.

50. (Previously Presented) The system of claim 48 further comprising recovering a storage device coupled with the computer by reading a portion of the block of data and writing the portion to the storage device using the pre-boot recovery utility.

51. (Previously Presented) The system of claim 48 further comprising recovering a corrupted operating system boot target stored on the storage device using the pre-boot recovery utility, wherein the medium includes a magnetic backup tape.